

Projektová dokumentace

Překladač jazyka IFJ22

Tým xstrel03 Varianta TRP

3. prosince 2022

Matyáš Strelec	(xstrel03)	X%
Ondřej Seidl	(xseidl06)	X%
Maxmilián Nový	(xnovym00)	X%
Dominik Klon	(xklond00)	X%

Obsah

1	Práce v týmu	2
	1.1 Rozdělení práce	2
	1.2 Odchylky od rozvnoměrného rozdělení	2
2	Lexikální analýza	3
	2.1 Datové struktury	3
	2.2 Funkce	3
	2.3 Diagram konečného automatu	3
3	Syntaktická analýza	5
	3.1 Implementace	5
	3.2 LL-gramatika	5
	3.3 LL-tabulka	5
	3.4 Precedenční tabulka	5
4	Sémantická analýza	5
5	Tabulka symbolů	5
6	Generování kódu	5
7	Zpracování výrazů	5
8	Chybové hlášení	5
9	Hlavní program	5

1 Práce v týmu

Rozdělení práce mezi členy týmu (uveď te kdo a jak se podílel na jednotlivých částech projektu; povinně zdůvodněte odchylky od rovnoměrného rozdělení bodů).

1.1 Rozdělení práce

Matyáš Strelec

- Lexikální analýza
- Syntaktická analýzy
- Dokumentace

Ondřej Seidl

- Implementace tabulky symbolů
- Zpracování výrazů

Maxmilián Nový

- Návrh LL-gramatiky
- Syntaktická analýza

Dominik Klon

• Generování kódu

1.2 Odchylky od rozvnoměrného rozdělení

2 Lexikální analýza

2.1 Datové struktury

Implementace lexikální analýzy je obsažena v souborech lexer.c a lexer.h. Pro potřeby lexikálního analyzátoru byly vytvořeny datové struktury které pomáhají při práci s tokeny a konečným automatem. Výčtový typ fsm_state_t obsahuje všechny možné stavy konečného automatu dle návrhu, výčtový typ token_type_t definuje typy tokenů.

Struktura token_t obsahuje informace o tokenu, jeho typ, pozici v souboru, délku, a jeho předchůdce a následníka ve spojovém seznamu. Struktura token_list_t obsahuje ukazatele na první, poslední, a aktuální token.

2.2 Funkce

Všechny funkce jsou ve zdrojových souborech popsané v komentářích, včetně jejich funkcionality, parametrů a návratových hodnot.

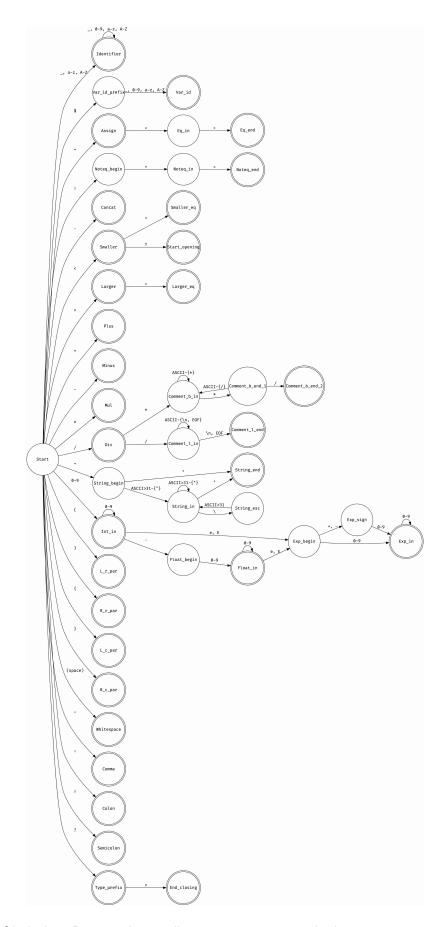
Funkce lexeru volaná z hlavního programu je funkce fillTokenList(), která jako parametr dostává ukazatel na strukturu token_list_t, kterou naplní seznamem tokenů pomocí volání funkce getNextToken(). Funkce getNextToken() je volána v cyklu, dokud není dosažen token typu konec souboru.

Funkce getNextToken() je implementována pomocí konečného automatu. Dle posloupnosti znaků na vstupu určuje typ a vyplňuje data tokenu. V případě, že je na vstupu znak, který nelze podle automatu dále číst, je kontrolováno, jestli momentální stav automatu je koncový, pokud ano, token je validní. Dále jsou rozpoznána klíčová slova a odstraněny úvozovky z řetězců. Pokud automat není v koncovém stavu, ale na vstup přijde znak, který automat nemůže přečíst, funkce vrací chybu 1.

Dále soubor obsahuje funkce na práci se seznamem tokenů jako vázaným seznamem a funkce pro ladění.

2.3 Diagram konečného automatu

Vizte obrázek 1.



Obrázek 1: Diagram konečného automatu vytvořený nástrojem Graphviz

- 3 Syntaktická analýza
- 3.1 Implementace
- 3.2 LL-gramatika
- 3.3 LL-tabulka
- 3.4 Precedenční tabulka
- 4 Sémantická analýza
- 5 Tabulka symbolů
- 6 Generování kódu
- 7 Zpracování výrazů
- 8 Chybové hlášení
- 9 Hlavní program